



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
[www.uspto.gov](http://www.uspto.gov)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/643,628	08/18/2003	Wei Li	50277-2250	4451
42425	7590	02/06/2008	EXAMINER	
HICKMAN PALERMO TRUONG & BECKER/ORACLE 2055 GATEWAY PLACE SUITE 550 SAN JOSE, CA 95110-1083			SAEED, USMAAN	
		ART UNIT	PAPER NUMBER	
		2166		
		MAIL DATE	DELIVERY MODE	
		02/06/2008	PAPER	

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

mn

<b>Office Action Summary</b>	<b>Application No.</b>	<b>Applicant(s)</b>	
	10/643,628	LI ET AL.	
	<b>Examiner</b>	<b>Art Unit</b>	
	Usmaan Saeed	2166	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

#### Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or-extended-period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

#### Status

- 1) Responsive to communication(s) filed on 31 October 2007.
- 2a) This action is FINAL.                    2b) This action is non-final.
- 3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

#### Disposition of Claims

- 4) Claim(s) 1-7,9-20 and 22-26 is/are pending in the application.
  - 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) Claim(s) \_\_\_\_\_ is/are allowed.
- 6) Claim(s) 1-7,9-20 and 22-26 is/are rejected.
- 7) Claim(s) \_\_\_\_\_ is/are objected to.
- 8) Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

#### Application Papers

- 9) The specification is objected to by the Examiner.
- 10) The drawing(s) filed on 18 August 2003 is/are: a) accepted or b) objected to by the Examiner.
 

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

#### Priority under 35 U.S.C. § 119

- 12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
  - a) All    b) Some \* c) None of:
    1. Certified copies of the priority documents have been received.
    2. Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
    3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

#### Attachment(s)

- |  |   |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)          | 4) <input type="checkbox"/> Interview Summary (PTO-413)           |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____.                                     |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)          | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| Paper No(s)/Mail Date _____.   | 6) <input type="checkbox"/> Other: _____.                         |

## DETAILED ACTION

### ***Continued Examination Under 37 CFR 1.114***

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 10/31/2007 has been entered.

### ***Claim Rejections - 35 USC § 103***

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

Claims 1-2, 4-7, 12-15, 17-20, and 25-26 are rejected under 35 U.S.C. 103(a) as being unpatentable over **Agrawal et al.** (**Agrawal** hereinafter) (U.S. Patent No. 6,324,533) in view of **Ching Chen et al.** (**Chen** hereinafter) (U.S PG Pub No. 2002/0087561).

With respect to claim 1, **Agrawal** teaches a method for performing a frequent itemset operation, the method comprising the steps of:

“within a database server that supports a particular database language, parsing a database statement to detect within the database statement a construct that extends the particular database language” as an object of the present invention is to provide a method for mining data relationships from the integrated mining system in the form of queries to SQL engines, and with k-way join, three-way join, subqueries, and group-by operations for counting the itemset support (**Agrawal** Col 2, Lines 27-31). A method for mining data relationships from the integrated mining system in the form of queries to SQL engines enhanced with object-relational extensions (SQL-OR), such as user-defined functions (UDFs) and table functions (**Agrawal** Col 2, Lines 33-36).

“wherein the construct identifies a function that counts and return frequent itemsets” as the group-by query preferably includes the steps of counting the number of transactions that contain each item and selecting the items that have a support above a user-specified threshold in determining the frequent itemsets (**Agrawal** Col 2, Lines 53-56).

**"performing said frequent itemset operation as part of execution of the database statement to produce results"** as the mining operation is expressed in some extension of SQL or a graphical language, which are input to preprocessor 21. This preprocessor generates appropriate SQL translations for the mining operation. For example, these SQL translations may be those that are executed by a SQL-92 relational engine 22. It is assumed that blobs, user-defined functions, and table functions are available in the object-relational engine. The mining results might be output to a depository 24 (**Agrawal** Col 6, Lines 26-42 and Figure 3).

**"storing the results in a computer-readable medium"** as figure 1 reference numeral 9 (**Agrawal** Figure 1).

**Agrawal** teaches the elements of claim 1 as noted above but does not explicitly teaches **"a cursor as input."**

However, **Chen** teaches **"a cursor as input"** as control begins at block 200 with the executive 6 receiving an OPEN command for a static cursor scroll. The DECLARE statement for the static scrollable cursor would have been previously processed. The executive 6 then calls (at block 202) the parser compiler 8 and optimizer 10 to parse and optimize the OPEN statement. After the OPEN statement is parsed and optimized, the executive 6 calls (at block 204) the structure generator 12 to construct an INSERT command from the SELECT statement in the previously compiled and executed DECLARE statement to populate the rows of the result table 50 with the qualifying rows of the base table 60 (**Chen** Paragraph 0051).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaching of the cited references because **Chen's** teachings would have allowed **Agrawal** to provide high concurrency for the rows in the base table when cursors are used by obtaining a lock on the rows in the base table for the duration of the cursor operation.

With respect to claim 2, **Agrawal** teaches "**the method of claim 1, wherein the database statement is expressed in a particular database language, and wherein the particular database language is SQL**" as a method for mining data relationships from the integrated mining system in the form of queries to SQL engines enhanced with object-relational extensions (SQL-OR), such as user-defined functions (UDFs) and table functions (**Agrawal** Col 2, Lines 33-36).

With respect to claim 4, **Agrawal** teaches **the method of claim 1 wherein: "the database statement includes a first indication of a first input format"** as the data table is first transformed into a vertical format by creating for each item a BLOB containing all tids that contain that item (Tid-list creation phase) and then count the support of itemsets by merging together these tid-lists (support counting phase) (**Agrawal** Col 12, Lines 43-47).

**"the frequent itemset operation operates on input that conforms to said first input format"** as a table function Gather is used for creating the Tid-lists. This is

the same as the Gather function in GatherJoin except here, the tid-list is created for each frequent item. The data table T is scanned in the (item, tid) order and passed to the function Gather. The function collects the tids of all tuples of T with the same item in memory and outputs a (item, tid-list) tuple for items that meet the minimum support criterion. The tid-lists are represented as BLOBs and stored in a new TidTable with attributes (item, tid-list) (**Agrawal Col 12, Lines 48-56**).

**"the method further comprises the steps of: parsing a second database statement to detect within the second database statement the construct that extends a database language"** as a method for mining data in an integrated database and data-mining system. Start with step 30, a group-by query is performed on the data transactions to generate a set of frequent 1-itemsets. One-itemsets are those having exactly one item each, while an itemset is frequent if the number of transactions containing it is at least at a specified number. At step 31, frequent 2-itemsets are determined from the frequent 1-itemsets and the transaction table. A candidate set of (n+2)-itemsets is next generated in step 32 from the frequent (n+1)-itemsets, where n=1. At step 33, frequent (n+2)-itemsets are generated from the candidate set of (n+2)-itemsets and the transaction table using a query (**Agrawal Col 6, Lines 43-55**). A first query is being performed to generate 1-itemsets, and (n+2) itemsets are being generated using another query. **"wherein the second database statement includes a second indication of a second input format that is different from said first input format"** as a horizontal format where each tid is followed by a collection of all its items (**Agrawal Col 10, Lines 37-38**).

**"in response to detection of said construct in said second database statement, the database server performing a second frequent itemset operation as part of execution of the second database statement"** as the mining operation is expressed in some extension of SQL or a graphical language, which are input to preprocessor 21. This preprocessor generates appropriate SQL translations for the mining operation. For example, these SQL translations may be those that are executed by a SQL-92 relational engine 22. It is assumed that blobs, user-defined functions, and table functions are available in the object-relational engine. The mining results might be output to a depository 24 (**Agrawal Col 6, Lines 26-42**). **"wherein the second frequent itemset operation operates on input that conforms to said second format"** as K-way Join approach where the k-way self join of T is replaced with the table functions Gather and Comb-K. It is possible to merge these functions together as a single table function GatherComb-K. The Gather function is not required when the data is already in a horizontal format where each tid is followed by a collection of all its items (**Agrawal Col 10, Lines 33-38**).

With respect to claim 5, **Agrawal** teaches **"the method of claim 4 wherein the first indication is identification of a first table function"** as a table function Gather is used for creating the Tid-lists. This is the same as the Gather function in GatherJoin except here, the tid-list is created for each frequent item. The data table T is scanned in the (item, tid) order and passed to the function Gather. The function collects the tids of all tuples of T with the same item in memory and outputs a (item, tid-list) tuple for items

that meet the minimum support criterion (**Agrawal** Col 12, Lines 48-56). “**and the second indication is identification of a second table function**” as the output of Gather is passed to another table function Comb-K which returns all k-item combinations formed out of the items of a transaction (**Agrawal** Col 10, Lines 24-27).

With respect to claim 6, **Agrawal** teaches “**the method of claim 1 wherein the frequent itemset operation uses, as input, a row source that is generated during execution of other operations specified in said database statement**” as output is a collection of rules of varying length. The maximum length of these rules is much smaller than the number of items and is rarely more than a dozen. Therefore, a rule is represented as a tuple in a fixed-width table where the extra column values are set to NULL to accommodate rules involving smaller itemsets. The schema of a rule is (item.sub.1, . . . , item.sub.k, len, rulem, confidence, support) where k is the size of the largest frequent itemset (**Agrawal** Col 5, Lines 65-67 & Col 6, Lines 1-6). A table function, GenRules, is used to generate all possible rules from a frequent itemset. The input to the function is a frequent itemset. For each itemset, it outputs tuples corresponding to rules with all non-empty proper subsets of the itemset in the consequent. The table function outputs tuples with k+3 attributes, T\_item.sub.1, . . . , T\_item.sub.k, T\_support, T\_ten, T\_rulem (**Agrawal** Col 8, Lines 7-13). From first operation a row/tuple is being obtained, which is then being used as an input.

With respect to claim 7, **Agrawal** teaches “**the method of claim 1 wherein the frequent itemset operation produces, as output, a row source that is used as input for other operations specified in said database statement**” as output is a collection of rules of varying length. The maximum length of these rules is much smaller than the number of items and is rarely more than a dozen. Therefore, a rule is represented as a tuple in a fixed-width table where the extra column values are set to NULL to accommodate rules involving smaller itemsets. The schema of a rule is (item.sub.1, . . . , item.sub.k, len, rulem, confidence, support) where k is the size of the largest frequent itemset (**Agrawal** Col 5, Lines 65-67 & Col 6, Lines 1-6). A table function, GenRules, is used to generate all possible rules from a frequent itemset. The input to the function is a frequent itemset. For each itemset, it outputs tuples corresponding to rules with all non-empty proper subsets of the itemset in the consequent. The table function outputs tuples with k+3 attributes, T\_item.sub.1, . . . , T\_item.sub.k, T\_support, T\_ten, T\_rulem (**Agrawal** Col 8, Lines 7-13). From first operation a row/tuple is being obtained as an output, which is then being used as an input.

With respect to claim 12, **Agrawal** teaches “**the method of claim 1 wherein the step of performing the frequent itemset operation includes performing a frequent itemset operation whose results identify frequent itemsets, and for each of the frequent itemsets, a count of how many item groups included the frequent itemset**” as a set of frequent 1-itemsets is generated using a group-by query on data

transactions. From these frequent 1-itemsets and the transactions, frequent 2-itemsets are determined. A candidate set of (n+2)-itemsets are generated from the frequent 2-itemsets; where n=1. Frequent (n+2)-itemsets are determined from candidate set and the transaction table using a query operation (**Agrawal Abstract**).

With respect to claim 13, **Agrawal** teaches “**the method of claim 1 wherein the step of performing the frequent itemset operation includes performing a frequent itemset operation whose results identify frequent itemsets, and for each of the frequent itemsets, a count of how items are in the frequent itemset**” as a set of frequent 1-itemsets is generated using a group-by query on data transactions (**Agrawal Abstract**). The support counting phase, conceptually for each itemset in C.sub.k the tid-lists of all k items are collected and the number of tids in the intersection of these k lists is counted using a user defined function (UDF) (**Agrawal** Col 12, Lines 56-59).

3. Claims 3, 9-11, 16, and 22-24 are rejected under 35 U.S.C. 103(a) as being unpatentable over **Agrawal et al.** (U.S. Patent No. 6,324,533) in view of **Ching Chen et al.** (U.S PG Pub No. 2002/0087561) as applied to claim 1-2, 4-7, 12-15, 17-20, and 25-26 above further in view of **Roberto Javier Bayardo**. (**Bayardo** hereinafter) (U.S Patent No. 6,138,117).

With respect to claim 3, **Agrawal** teaches “**the method of claim 1, wherein the construct is a table function**” as a method for mining data relationships from the integrated mining system in the form of queries to SQL engines enhanced with object-relational extensions (SQL-OR), such as user-defined functions (UDFs) and table functions (**Agrawal** Col 2, Lines 33-36).

“**wherein the database statement specifies frequency criteria and additional criteria, wherein said frequency criteria specifies at least one criterion that relates to how frequently combination of items appear together**” as to find all combinations of items whose support is greater than minimum support. Call those combinations frequency itemsets (**Agrawal** Col 5, Lines 20-23).

“**wherein the results include frequent itemsets that satisfy both said frequency criteria and said additional criteria, and wherein the results do not include frequent itemsets that satisfy said frequency criteria but do not satisfy said additional criteria**” as the frequent (n+2)-itemsets are determined using cascaded subqueries by: a) selecting distinct first items in the candidate itemsets using a subquery (**Agrawal** Col 3, Lines 2-4). Using the results of the last subqueries to determine which of the (n+2)-itemsets are frequent. In generating rules from the union of the frequent itemsets, all items from the frequent itemsets are first put into a table F. A set of candidate rules is created from the table F using a table function. These candidate rules are joined with the table F, and filtered to remove those that do not meet a confidence criteria (**Agrawal** Col 3, Lines 9-16).

F consists of k+2 attributes (item.sub.1, . . . , item.sub.k, support, len), where k is the size of the largest frequent itemset and len is the length of the itemset (**Agrawal** Col 8, Lines 4-6). Sequence of operations can be implemented as a single SQL query for any k, as shown in FIG. 12. Therefore the query specifies both the frequency criteria and the additional criteria k, which is the size of an itemset.

**Agrawal** teaches the elements of claim 3 as noted above but does not explicitly disclose, “**wherein said additional criteria do not specify any criterion that related to how frequently combinations of items appear together.**”

However, **Bayardo** discloses, “**wherein said additional criteria do not specify any criterion that relates to how frequently combinations of items appear together**” as Max-Miner usually performs less database passes than this bound in practice when the longest frequent itemsets are more than 10 in length (**Bayardo** Col 9, Lines 57-60). Examiner interprets the length of 10 as additional criteria.

It is still another object of the present invention to quickly identify those patterns that are both frequent and maximal so that the set of maximal frequent patterns represents the set of all frequent patterns (**Bayardo** Col 3, Lines 32-35 and Lines 40-56).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaching of the cited references because **Bayardo's** teachings would have allowed **Agrawal and Chen** to provide an efficient method for extracting relatively long frequent patterns from a database of transaction records where each record includes several data items.

With respect to claim 9, **Agrawal** teaches “**the method of claim 1 wherein: the additional criteria specify a minimum length; and the step of performing the frequent itemset operation includes performing a frequent itemset operation whose results exclude all item sets that include fewer items than the minimum length specified by the additional criteria**” as combinations of items whose support is greater than minimum support. Call those combinations frequent itemsets (**Agrawal** Col 5, Lines 21-23). The function collects the tids of all tuples of T with the same item in memory and outputs a (item, tid-list) tuple for items that meet the minimum support criterion (**Agrawal** Col 12, Lines 52-55).

**Agrawal** further teaches the function collects the tids of all tuples of T with the same item in memory and outputs a (item, tid-list) tuple for items that meet the minimum support criterion. The tid-lists are represented as BLOBs and stored in a new TidTable with attributes (item, tid-list) (**Agrawal** Col 11, Lines 49-56).

**Agrawal** teaches the elements of claim 9 as noted above but does not explicitly teaches “**a minimum length.**”

However, **Bayardo** teaches “**a minimum length**” as Max-Miner usually performs less database passes than this bound in practice when the longest frequent itemsets are more than 10 in length (**Bayardo** Col 9, Lines 57-60). Examiner interprets the length of 10 as the minimum length.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaching of the cited references because

**Bayardo's** teachings would have allowed **Agrawal and Chen** to provide an efficient method for extracting relatively long frequent patterns from a database of transaction records where each record includes several data items.

With respect to claim 10, **Agrawal** teaches “**the method of claim 1 wherein: the additional criteria specify a maximum length; and the step of performing the frequent itemset operation includes performing a frequent itemset operation whose results exclude all item sets that include more items than the maximum length specified by the additional criteria**” as F consists of k+2 attributes (item.sub.1, . . . , item.sub.k, support, len), where k is the size of the largest frequent itemset and len is the length of the itemset (**Agrawal** Col 8, Lines 4-6).

**Agrawal** further teaches in particular, it is not practical to assume that all items in a transaction appear as different columns of a single tuple because often the number of items per transaction can be more than the maximum number of columns that the database supports. For instance, for one of our real-life datasets the maximum number of items per transaction is 872 and for another it is 700 (**Agrawal** Col 5, Lines 56-60).

**Agrawal** teaches the elements of claim 10 as noted above but does not explicitly teach “**a maximum length.**”

However, **Bayardo** discloses “**a maximum length**” as the most part, frequent-pattern mining methods have been developed to operate on databases in which the longest frequent patterns are relatively short, e.g., those with less than 10 items

(**Bayardo** Col 1, Lines 22-26). Examiner interprets the length of 10 as the maximum length.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaching of the cited references because **Bayardo's** teachings would have allowed **Agrawal and Chen** to provide an efficient method for extracting relatively long frequent patterns from a database of transaction records where each record includes several data items.

With respect to claim 11, **Agrawal** teaches “**the method of claim 1 wherein: the additional criteria specify a set of one or more included items; and the step of performing the frequent itemset operation includes performing a frequent itemset operation whose results exclude all itemsets that do not include all items in said set of one or more included items**” as the frequent (n+2)-itemsets are determined using cascaded subqueries by: a) selecting distinct first items in the candidate itemsets using a subquery. In generating rules from the union of the frequent itemsets, all items from the frequent itemsets are first put into a table F. These candidate rules are joined with the table F, and filtered to remove those that do not meet a confidence criteria (**Agrawal** Col 3, Lines 2-16).

**Agrawal** teaches the elements of claim 11 as noted above but does not explicitly teaches “**one or more included items.**”

However, **Bayardo** discloses “**one or more included items**” as a method for identifying patterns from a database of records including the steps of: (1) generating an initial set C of candidates where each candidate c includes two distinct sets of items: c.head and c.tail (**Bayardo** Col 3, Lines 42-45).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaching of the cited references because **Bayardo's** teachings would have allowed **Agrawal and Chen** to provide an efficient method for extracting relatively long frequent patterns from a database of transaction records where each record includes several data items.

#### ***Response to Arguments***

4. Applicant's arguments have been considered but are moot in view of the new ground(s) of rejection.

In these arguments applicant relies on amended claims and not the original ones.

See above rejections for response to the arguments.

Claims must be given the broadest reasonable interpretation during examination and limitations appearing in the specification but not recited in the claim are not read into the claim (See M.P.E.P. 2111 [R-I]).

***Contact Information***

5. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Usmaan Saeed whose telephone number is (571)272-4046. The examiner can normally be reached on M-F 8-5.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Hosain Alam can be reached on (571)272-3978. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Usmaan Saeed  
Patent Examiner  
Art Unit: 2166

  
Hosain Alam  
Supervisory Patent Examiner

US  
February 03, 2008